

REMARKS

In the Office Action mailed December 9, 2004, the Examiner noted that claims 1-17 were pending, and rejected claims 1-17. New claim 18 has been added and, thus, in view of the forgoing claims 1-18 remain pending for reconsideration which is requested. No new matter has been added. The Examiner's rejections are traversed below.

On page 2 of the Office Action, the Examiner rejected all claims under 35 U.S.C. § 102 as anticipated by Bates.

The present invention (see claims 1, 2, 11 and 15-18) is directed to a system for detecting a computer virus and providing infection information concerning the detected computer virus. The system particularly stores a communication (or execution) history of a terminal apparatus or client, specifies the time of infection based on the stored history when a computer virus is detected by installed anti-virus software, transmits the infection information including the specified time of infection from a central apparatus or server to the terminal apparatus and displays the transmitted infection information on the terminal apparatus.

In contrast, Bates discusses a web server performing "real time" virus checking of information as that information is requested by a web client (see Bates, col. 2, lines 31-34). This allows Bates to eliminate the need for virus checking software on the web client (see Bates, col. 2, lines 44-46).

On page 3 of the Action, the Examiner particularly asserts that Bates teaches the storage of a communication history and points to col. 5, line 57- col. 6, line 20 and col. 12, lines 37-58. This text of Bates particularly states:

The user list 127 is a list of users that are registered to use the virus control mechanism 131. The user list 127 includes a list of users, and their corresponding virus checking preferences 128 that determine how the web server application 123, e-mail server application 124, and/or virus control mechanism 131 screen incoming information for viruses. Virus control mechanism 131 includes the web page virus processing mechanism 132, e-mail virus processing mechanism 134, and file virus processing mechanism 136. Web page virus processing mechanism 132 checks a web client's request for a web page to determine whether the web page or any contained links were the source of a virus in the past. The virus information database 138 is a database of virus information that relates to web server computer system 100. Note that virus information database 138 may be a local database, or may be a large centralized database that includes the virus information for many web servers, such as a centralized database that could be accessed via a web site. Virus information database 138 may include a specification of known viruses, along with statistics for which ones have been encountered and when. In addition, virus information database 138 may include a list of web sites that are known to contain viruses, or from where viruses were downloaded. A web site that contains a virus or from which a virus was

downloaded is referred to herein as a "bad" URL. Using the virus information database 138, web page virus processing mechanism 132 can warn a web client that has requested a web page at a bad URL, or that has requested a web page that includes links to a bad URL.

(See Bates, col. 5, line 57- col. 6, line 20)

One of the significant features of the preferred embodiments is the presence of a virus information database 138, as shown in FIGS. 1 and 3. Note that the virus information database 138 is different than the virus definitions 126 used by the virus checker application 125. Virus information database 138 is a repository of information concerning detected viruses, including the number of times the virus was detected, the time of detection, the origin of the virus, etc. Having this information available to the web server computer system allows the web server to log virus-related information and perform analysis on that information as needed. For example, when a virus in an e-mail is detected, the sender of the e-mail may be recorded in the virus information database 138. If the user sends a virus a second time, the user may be labeled in the virus information database as a user that has a history of sending viruses. This could result in the sender being notified that the web server is not accepting e-mails from the sender for a period of time due to excessive e-mails with viruses. The sender can thus be "branded" as a source of viruses, allowing the web server computer system to take any suitable action based on that knowledge.

(See Bates, col. 6, line 20 and col. 12, lines 37-58)

While Bates does discuss a database of virus information including when a virus has been encountered and the time of detection, the discussion does not indicate that history of the terminal communication is stored in this database. The present invention particularly stores this "communication history" (see claims 1, 2, 11 and 15-18). This communication history is used to determine ("based on the stored communication history") the "time of infection" (see claims 1, 2, 11 and 15-18). The time of infection is different from the time of detection (or "time of find-out" - claim 3) as an infection can occur long before a virus is detected. The communication history allows the determination of the time of infection. The time of infection beneficially allows a more accurate determination of what other systems might have been infected before the virus was detected.

The Examiner, on page 3, also asserts that Bates teaches specifying the time of infection and points to col. 8, lines 1-22 and col. 9, lines 38-60). This text particularly states:

Referring now to FIG. 4, a method 400 in accordance with the preferred embodiments allows a virus checker on a web server to automatically check e-mail messages, web pages, and downloaded files for viruses before passing these on to a web client. Method 400 begins when a web client requests information that normally would flow through the web server to the web client (step 410). If the request does not require virus checking (step 420=NO), the requested information is sent to the web client (step 480). If the request requires virus checking (step 420=YES), a virus check is performed on the requested information (step 430). If no virus is found (step 440=NO), the requested information is sent to the web client (step 480). If a virus is found (step 440=YES), the web client is notified of the virus (step 450), and an entry is made in the virus information database (step 460) regarding the name of the virus, type, when

detected, etc. Finally, the appropriate authorities may be notified of the virus (step 470). The term "appropriate authorities" is a broad term that encompasses anyone who may need to know about the occurrence of a virus, including a network administrator of a local area network, a web site administrator, a contact person in a virus detection company, and appropriate law enforcement officials, such as local, state, federal, and international law enforcement agencies.

(See Bates, col. 7, line 66 - col. 8, line 22, inclusive of col. 8, lines 1-22)

Referring back to FIGS. 1 and 3, each of mechanisms 132, 134 and 136 perform different functions. One suitable method in accordance with the preferred embodiments for the e-mail processing mechanism 134 is illustrated as method 700 in FIG. 7. Method 700 begins when an e-mail message is received that is intended for one of the users in the user list 127 (step 710). If the virus checking of e-mail messages is not enabled in the user virus checking preferences 128 for the user that is the intended recipient of the e-mail message (step 712=NO), the e-mail is sent to the recipient (step 714). On the other hand, if virus checking of e-mail messages is enabled in the user virus checking preferences 128 for the intended recipient (step 712=YES), the e-mail virus processing mechanism reads the e-mail message (step 720), and checks the e-mail message body for viruses (step 722) using the selected virus checker application. Note that the term "e-mail message body" includes all parts of the e-mail other than attachments, including the fields for sender and recipient, subject line, main portion of message, etc. If no viruses are found (step 724=NO), and there are no attachments to the e-mail message (step 740=NO), the e-mail message is sent to the recipient (step 714). If a virus is found (step 724=YES), the e-mail message is deleted (step 730), and a separate e-mail is sent to the intended recipient of the e-mail informing the recipient that the deleted e-mail message contained a virus and was automatically deleted (step 732). In addition, any other information regarding the virus-infected e-mail message could be sent to the intended recipient in step 732 as well. Next, method 700 e-mails the sender of the e-mail message that included the virus to inform the sender that they sent a virus (step 734). This step is particularly significant because it prevents a user from repeatedly and unknowingly sending out a virus as part of an e-mail message. Next, information regarding the virus is entered into the virus information database (step 736). If this is the first time this web server has detected this particular virus, step 736 preferably makes a new entry in virus information database 138 with pertinent information regarding the virus. If the web server has seen this particular virus before, step 736 preferably updates an existing entry in virus information database 138. Note that the information in virus information database 138 may include any pertinent information regarding the virus including, without limitation, its size in bytes, where the virus came from, when the virus was detected, the location of each detection, etc. Next, method 700 notifies the appropriate authorities regarding the virus (step 738). As stated above, the authorities notified can include any human being or computer that has a need to know about computer viruses.

(See Bates, col. 9, lines 12-60, inclusive of lines 38-60)

This text discusses conditional virus checking, recording the detection of a new virus as a new database entry and letting the sender of an infected know about the virus. This text does not discuss specifying the "time of infection" (see claims 1, 2, 11 and 15-18).

It is submitted that the present claimed invention of the independent claims patentably distinguishes over Bates and withdrawal of the rejection is requested.

The dependent claims depend from the above-discussed independent claims and are patentable over the prior art for the reasons discussed above. The dependent claims also recite additional features not taught or suggested by the prior art. For example, claim 3 calls for also storing the time of installation of the anti-virus software and using that information along with the time of detection (find-out) and the communication history to specify the time of infection. It is submitted that the dependent claims are independently patentable over the prior art.


It is submitted that the claims are not taught, disclosed or suggested by the prior art. The claims are therefore in a condition suitable for allowance. An early Notice of Allowance is requested.

If any further fees, other than and except for the issue fee, are necessary with respect to this paper, the U.S.P.T.O. is requested to obtain the same from deposit account number 19-3935.

Respectfully submitted,

STAAS & HALSEY LLP

Date: 5/6/15

By: 
J. Randall Beckers
Registration No. 30,358

1201 New York Avenue, NW, Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501